



MIAOW: An Open Source GPGPU

www.miaowgpu.org

Vinay Gangadhar, Raghu Balasubramaniam, Mario Drumond, Ziliang Guo,
Jai Menon, Cherin Joseph, Robin Prakash, Sharath Prasad, Pradip Vallathol,
Karu Sankaralingam

Vertical Research Group
UW - Madison

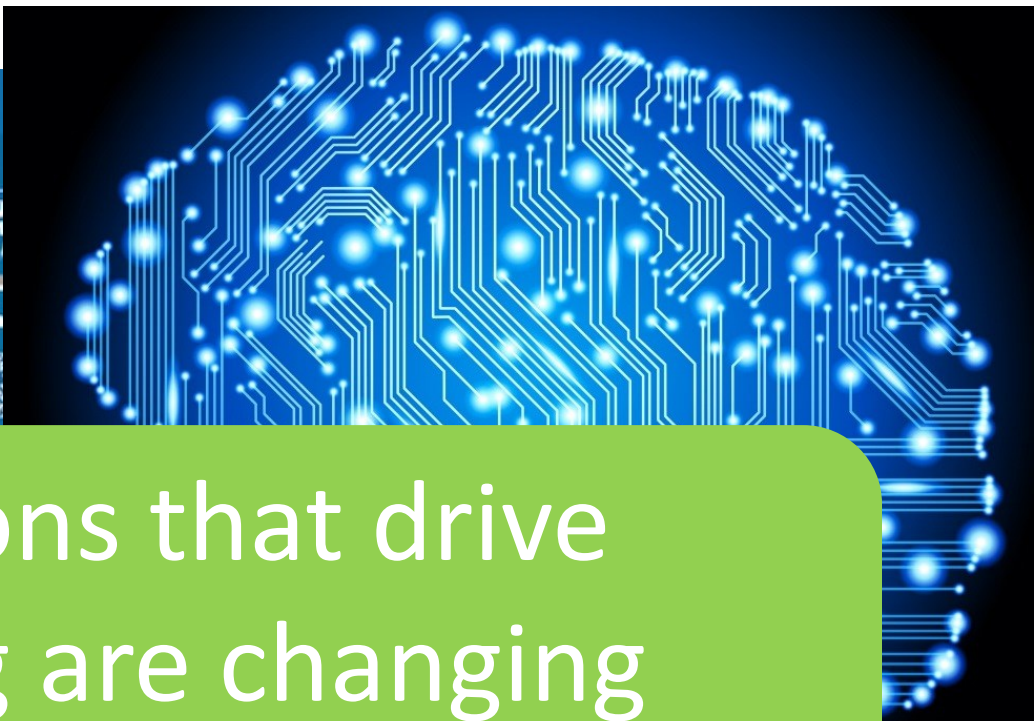




Executive Summary

- MIAOW is a **credible GPGPU implementation**
 - Compatible with AMD Southern Islands ISA
 - Runs OpenCL programs and prototyped on FPGA
 - Similar design to industry state-of-art
 - Similar performance to industry state-of-art*
 - Flexible and Extendable
- MIOAW's hardware design is **Open Source**
- Contributes to changing hardware landscape





Applications that drive
computing are changing
Need innovative new hardware

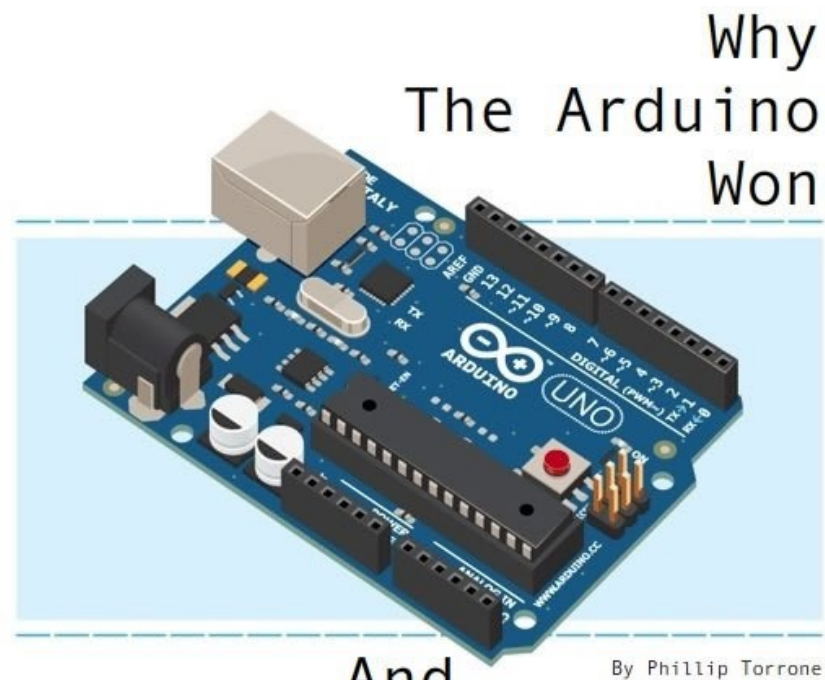




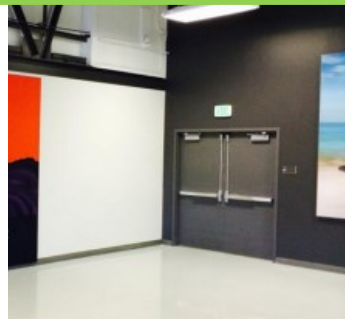
131 maker fairs
750K people



Open Source Hardware
gaining momentum



And
Why it's
Here to Stay



[View more photos and videos](#)



OPEN
Compute Project



Some Open Source Hardware Microprocessors



ZERO Open Source GPUs



Lessons from Open Source S/W

PHP, Linux, ruby, mysql,
sqlite, apache, gcc
late 80s, early 90s

\$0

Facebook, Twitter,
Whatsapp, Instagram
Web 2.0

>\$10 bill.

MIAOW, OpenCores,
RISC-V etc..

\$0

?



MIAOW Technical Overview

Demonstrate MIAOW is credible
GPGPU

Implications and Possibility of
Open Source Hardware



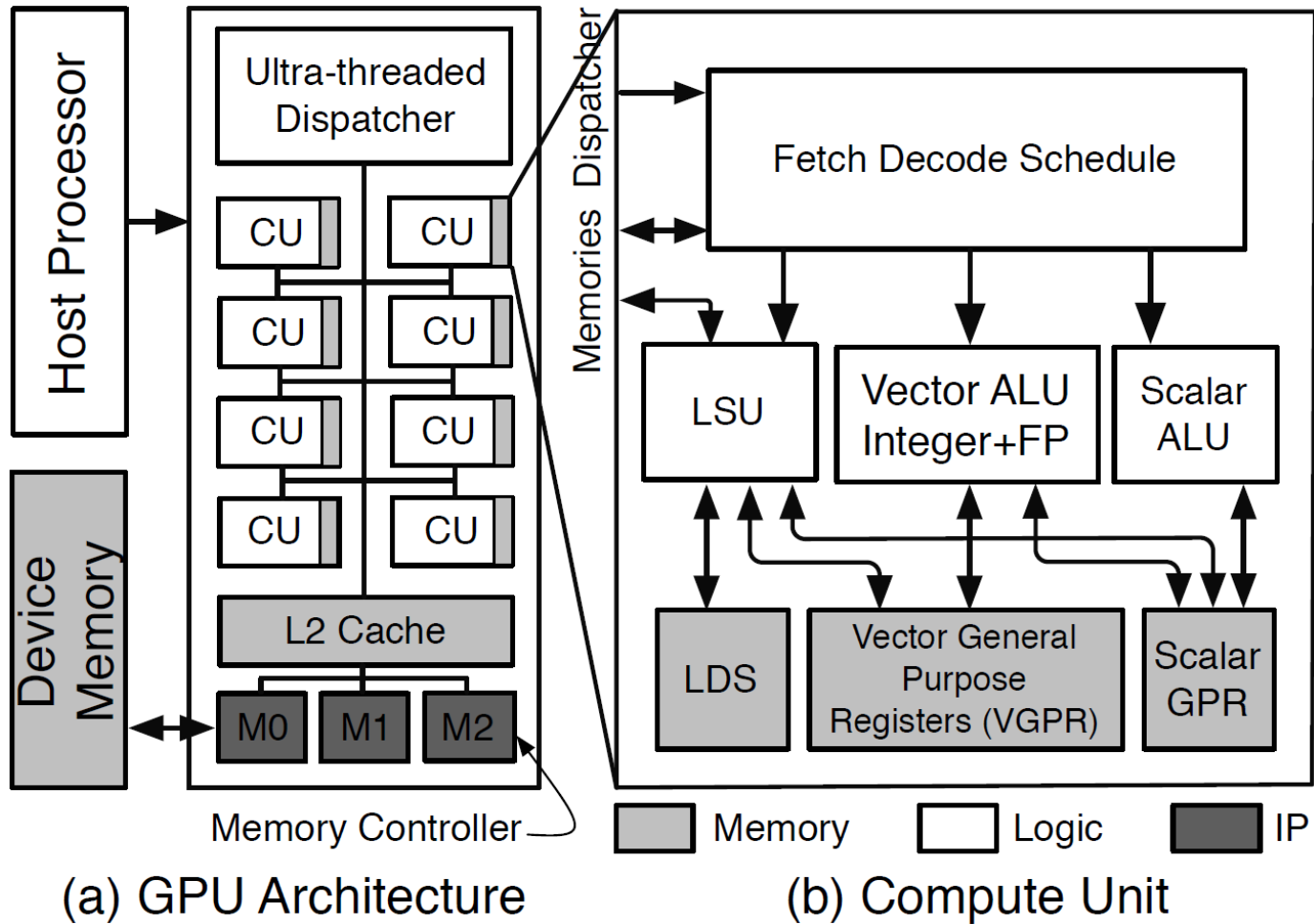
ISA Summary

Type	Instructions	
Vector	ALU:	add, addc, sub, mad, madmk, mac, mul, max, max3, min, subrev
	Bitwise:	and, or, xor, not, mov, lshrrev, lshlrev, ashrev, ashrrrev, bfe, bfi, cndmask
	Compare:	cmp_{ lt, eq, le, gt, lg, ge, ne, ng, neq }
Scalar	ALU:	add, addk, sub, max, min, mul, mulk
	Bitwise:	and, andn2, or, xor, not, mov, movk, lshl, lshr, ash, saveexec
	Compare:	cmp_{ eq, lt, gt, ge, lt, le, eq, lg, gt, ge, lt, le }
	Conditional:	barrier, branch, cbranch, endpgm, waitcnt
Memory	Scalar_Mem:	load, buffer_load
	Vector_Mem:	tbuffer_load, tbuffer_store
	Date Share (LDS, GDS):	ds_read, ds_write

- 95 instructions
- Single-precision support only
- No graphics support (yet)



MIAOW Overview

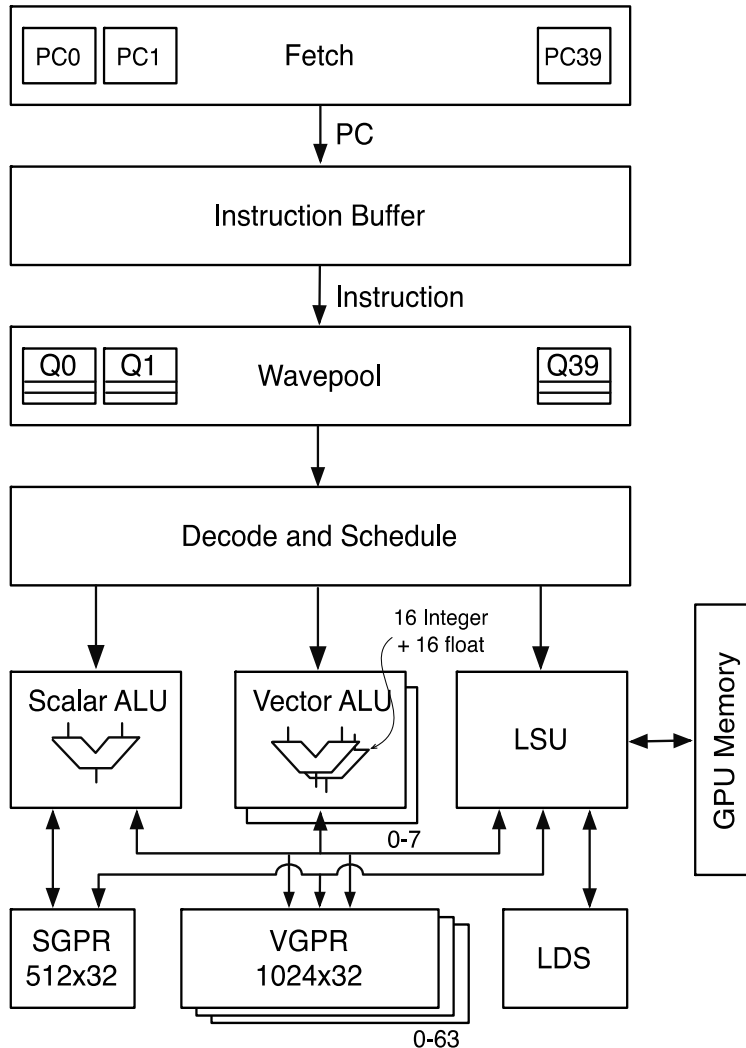


MIAOW has **32** Compute Units (CUs)



Hardware Organization

CU Microarchitecture



- Single Issue

- 40

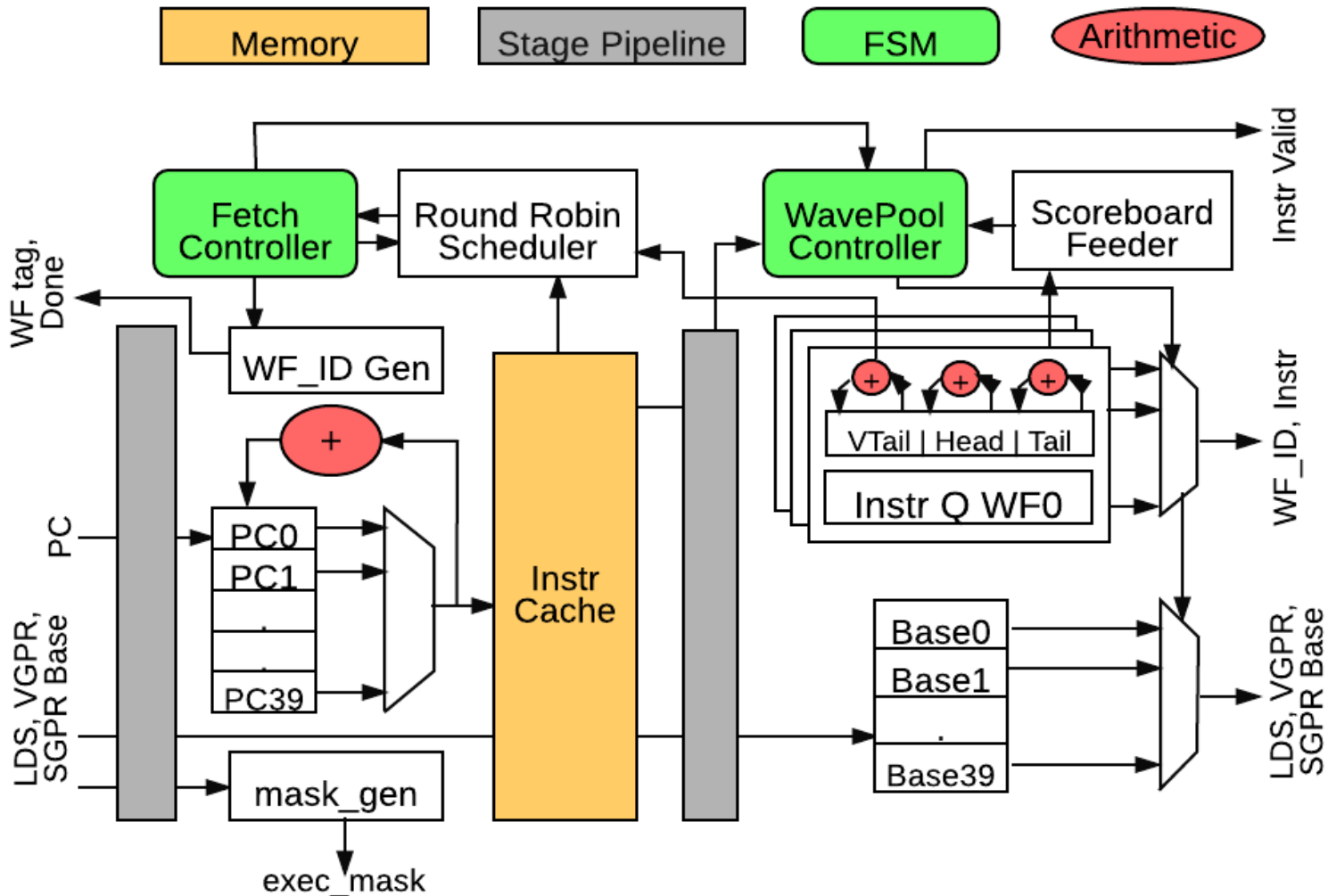


Wavefronts

- 16-wide vector ALUs
- LSU – Memory operations



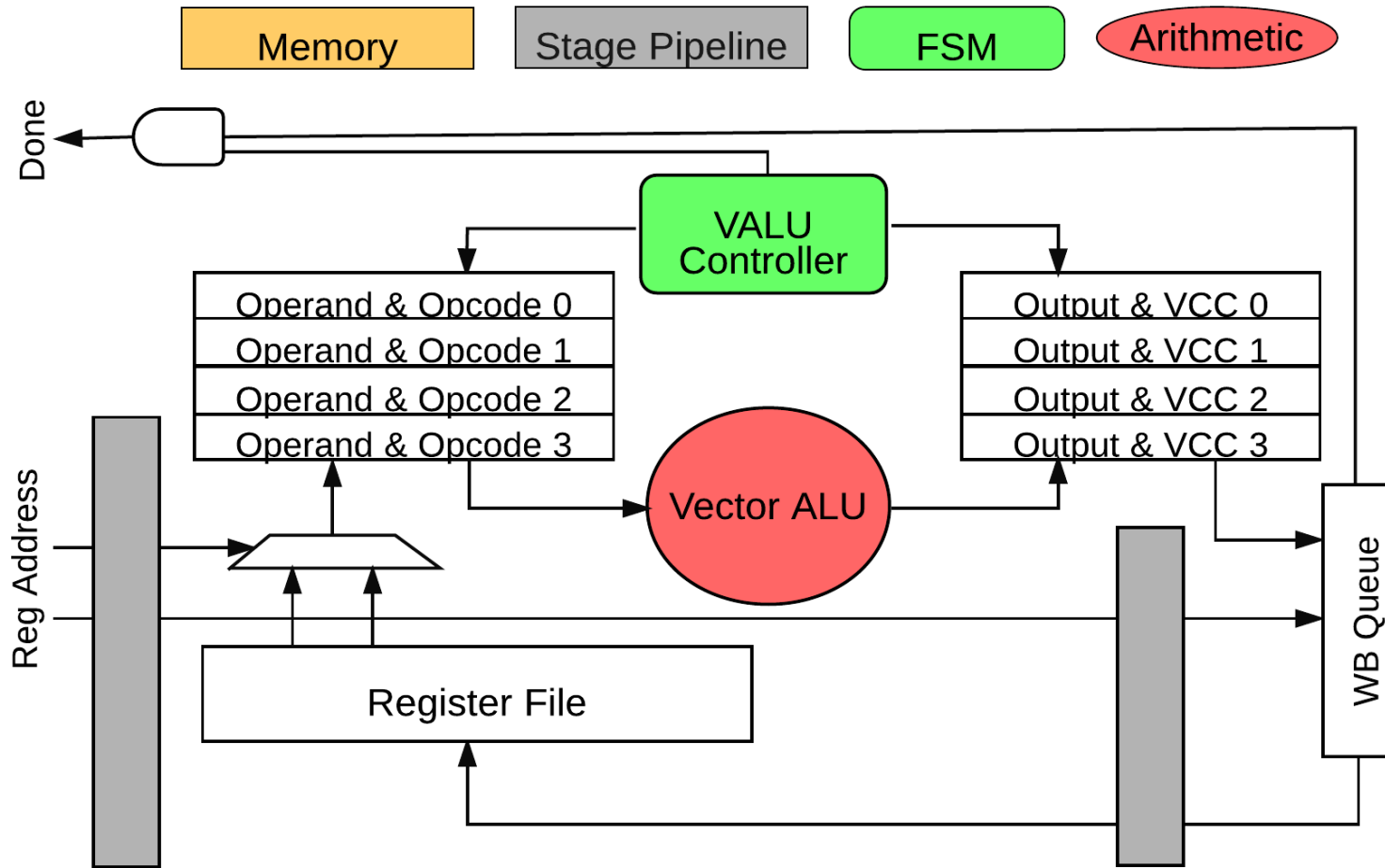
Fetch & Wavepool



Fetch & Wavepool



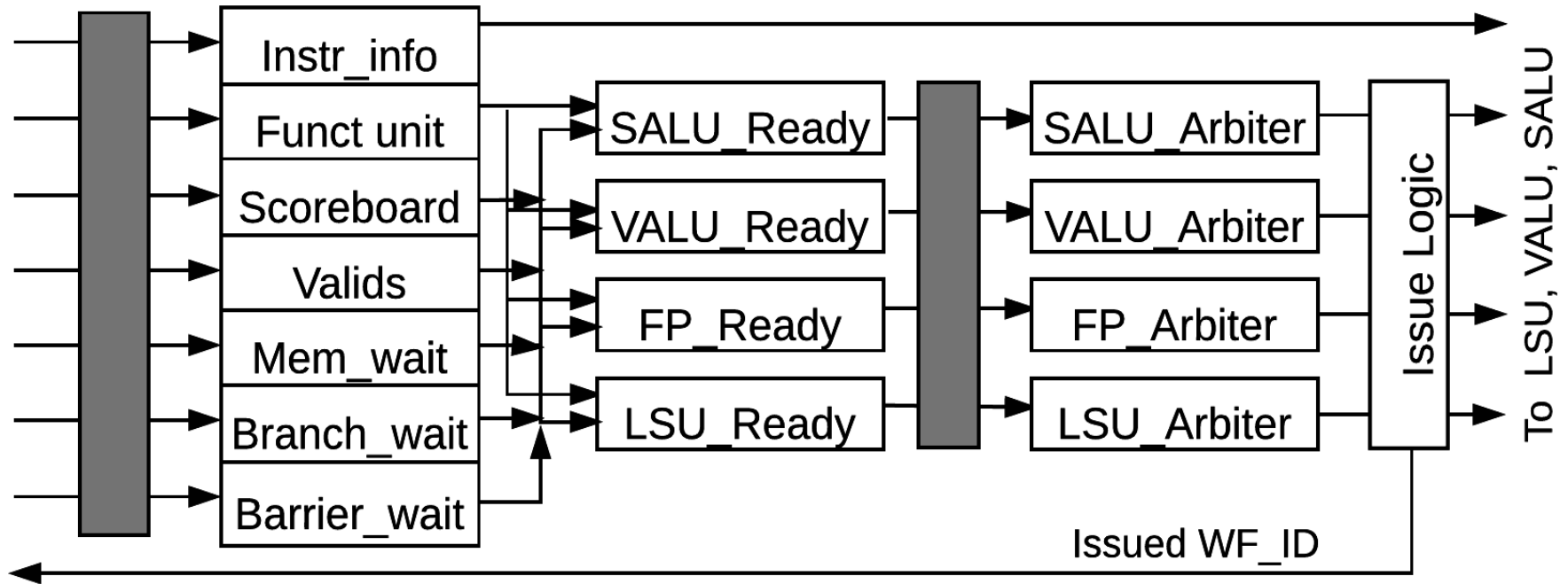
Vector ALU/FPU



Vector ALU/Vector FPU



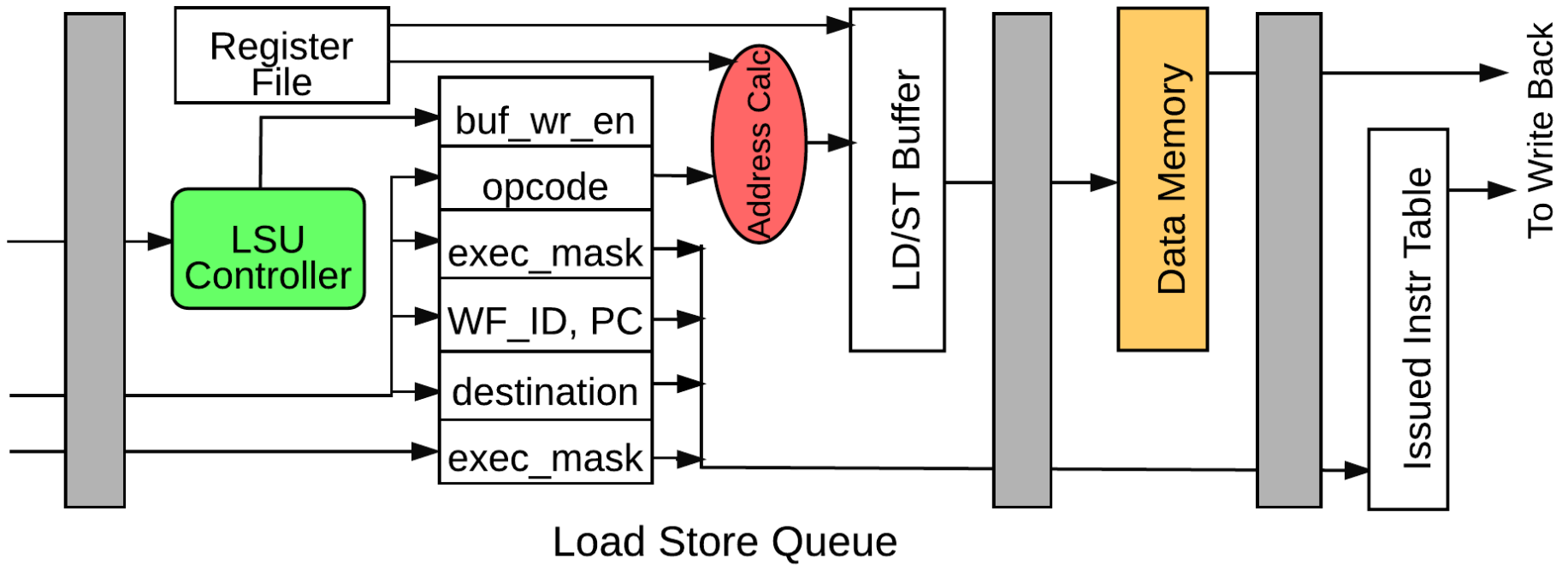
Issue



Issue: Wavefront Scheduling and Arbiter



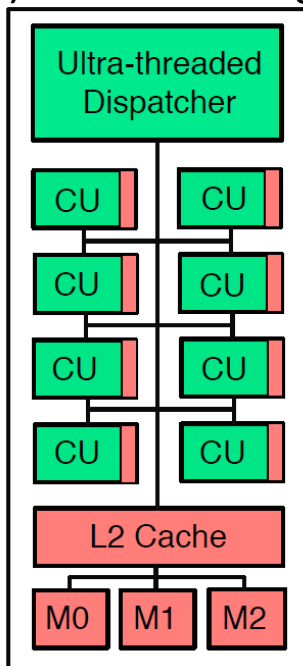
Load/Store Unit



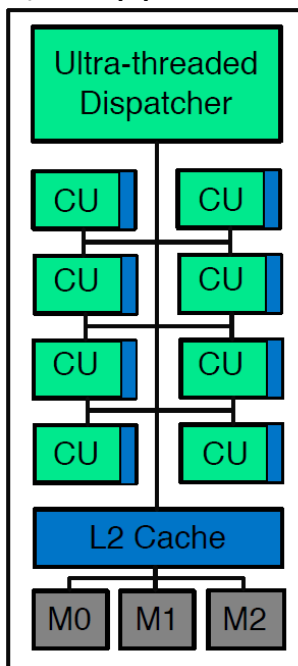


MIAOW Implementations

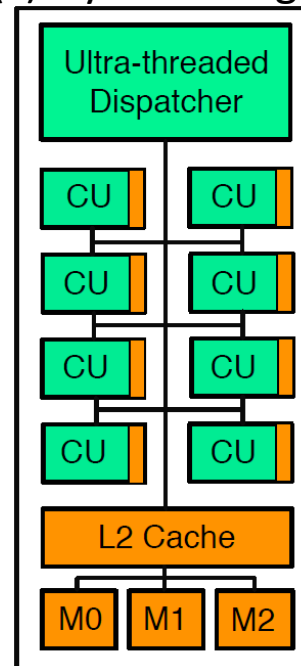
(a) Full ASIC Design



(b) Mapped to FPGA



(c) Hybrid Design



Legend

CU : Compute Unit; M0, M1, M2 : Memory Controllers



Low Flexibility



High Realism



High Cost

Medium Flexibility

Medium Realism



Low Cost



Long Design Time

High Flexibility

Flexible Realism

Low Cost

Short Design Time





- The figure displays a 3x3 grid of FPGA resource utilization plots for a 301497 device. Each plot shows the distribution of resources (ALUs, registers, LUTs, etc.) for different components. The plots are arranged in a 3x3 grid, with the top row showing 'simf0', 'issue0', and 'scoreboard', the middle row showing 'simd0', and the bottom row showing 'wavepool0'. Each plot includes a title, a numerical value, and a detailed resource utilization breakdown.

Component	Value	Resources
simf0	103050	alu: 53792, val: 103050, wb_queue: 19867, queue_reg: 19845, src_shift_reg: 19660, alu_controller: 8708, Leaf Cells: 8686
issue0	45376	scoreboard: 27395, 17266, 2228, 2204
simd0	75912	src_shift_reg: 33432, wb_queue: 19930, queue_reg: 19900, src1_shift: 12071, src2_shift: 8780, alu_controller: 13008, Leaf Cells: 12080
wavepool0	32584	pool: 30842, fetch0: 12468, exec0: 9200, vacant_reg: 10172, writeback: 7569, m0: 10172, m1: 10172, m2: 10172, m3: 10172, m4: 10172, m5: 10172, m6: 10172, m7: 10172, m8: 10172, m9: 10172, m10: 10172, m11: 10172, m12: 10172, m13: 10172, m14: 10172, m15: 10172, m16: 10172, m17: 10172, m18: 10172, m19: 10172, m20: 10172, m21: 10172, m22: 10172, m23: 10172, m24: 10172, m25: 10172, m26: 10172, m27: 10172, m28: 10172, m29: 10172, m30: 10172, m31: 10172, m32: 10172, m33: 10172, m34: 10172, m35: 10172, m36: 10172, m37: 10172, m38: 10172, m39: 10172, m40: 10172, m41: 10172, m42: 10172, m43: 10172, m44: 10172, m45: 10172, m46: 10172, m47: 10172, m48: 10172, m49: 10172, m50: 10172, m51: 10172, m52: 10172, m53: 10172, m54: 10172, m55: 10172, m56: 10172, m57: 10172, m58: 10172, m59: 10172, m60: 10172, m61: 10172, m62: 10172, m63: 10172, m64: 10172, m65: 10172, m66: 10172, m67: 10172, m68: 10172, m69: 10172, m70: 10172, m71: 10172, m72: 10172, m73: 10172, m74: 10172, m75: 10172, m76: 10172, m77: 10172, m78: 10172, m79: 10172, m80: 10172, m81: 10172, m82: 10172, m83: 10172, m84: 10172, m85: 10172, m86: 10172, m87: 10172, m88: 10172, m89: 10172, m90: 10172, m91: 10172, m92: 10172, m93: 10172, m94: 10172, m95: 10172, m96: 10172, m97: 10172, m98: 10172, m99: 10172, m100: 10172, m101: 10172, m102: 10172, m103: 10172, m104: 10172, m105: 10172, m106: 10172, m107: 10172, m108: 10172, m109: 10172, m110: 10172, m111: 10172, m112: 10172, m113: 10172, m114: 10172, m115: 10172, m116: 10172, m117: 10172, m118: 10172, m119: 10172, m120: 10172, m121: 10172, m122: 10172, m123: 10172, m124: 10172, m125: 10172, m126: 10172, m127: 10172, m128: 10172, m129: 10172, m130: 10172, m131: 10172, m132: 10172, m133: 10172, m134: 10172, m135: 10172, m136: 10172, m137: 10172, m138: 10172, m139: 10172, m140: 10172, m141: 10172, m142: 10172, m143: 10172, m144: 10172, m145: 10172, m146: 10172, m147: 10172, m148: 10172, m149: 10172, m150: 10172, m151: 10172, m152: 10172, m153: 10172, m154: 10172, m155: 10172, m156: 10172, m157: 10172, m158: 10172, m159: 10172, m160: 10172, m161: 10172, m162: 10172, m163: 10172, m164: 10172, m165: 10172, m166: 10172, m167: 10172, m168: 10172, m169: 10172, m170: 10172, m171: 10172, m172: 10172, m173: 10172, m174: 10172, m175: 10172, m176: 10172, m177: 10172, m178: 10172, m179: 10172, m180: 10172, m181: 10172, m182: 10172, m183: 10172, m184: 10172, m185: 10172, m186: 10172, m187: 10172, m188: 10172, m189: 10172, m190: 10172, m191: 10172, m192: 10172, m193: 10172, m194: 10172, m195: 10172, m196: 10172, m197: 10172, m198: 10172, m199: 10172, m200: 10172, m201: 10172, m202: 10172, m203: 10172, m204: 10172, m205: 10172, m206: 10172, m207: 10172, m208: 10172, m209: 10172, m210: 10172, m211: 10172, m212: 10172, m213: 10172, m214: 10172, m215: 10172, m216: 10172, m217: 10172, m218: 10172, m219: 10172, m220: 10172, m221: 10172, m222: 10172, m223: 10172, m224: 10172, m225: 10172, m226: 10172, m227: 10172, m228: 10172, m229: 10172, m230: 10172, m231: 10172, m232: 10172, m233: 10172, m234: 10172, m235: 10172, m236: 10172, m237: 10172, m238: 10172, m239: 10172, m240: 10172, m241: 10172, m242: 10172, m243: 10172, m244: 10172, m245: 10172, m246: 10172, m247: 10172, m248: 10172, m249: 10172, m250: 10172, m251: 10172, m252: 10172, m253: 10172, m254: 10172, m255: 10172, m256: 10172, m257: 10172, m258: 10172, m259: 10172, m260: 10172, m261: 10172, m262: 10172, m263: 10172, m264: 10172, m265: 10172, m266: 10172, m267: 10172, m268: 10172, m269: 10172, m270: 10172, m271: 10172, m272: 10172, m273: 10172, m274: 10172, m275: 10172, m276: 10172, m277: 10172, m278: 10172, m279: 10172, m280: 10172, m281: 10172, m282: 10172, m283: 10172, m284: 10172, m285: 10172, m286: 10172, m287: 10172, m288: 10172, m289: 10172, m290: 10172, m291: 10172, m292: 10172, m293: 10172, m294: 10172, m295: 10172, m296: 10172, m297: 10172, m298: 10172, m299



Design Team

- Small initial design team (12 mo)
 - 5-person HDL team
 - 1-person software team
 - 1-person physical design team
- Added FPGA expert
- 3 undergrads extended the design
- Total duration: 36 months
- Area, Frequency, Performance, Power – NON GOALS



Software Compatibility

- Runs unmodified OpenCL programs
- All **AMD APP SDK** OpenCL benchmarks
- Many Rodinia benchmarks
- Easily extendable to add additional instructions



MIAOW Technical Overview

**Demonstrate MIAOW is a
credible GPGPU**

Implications and Possibility of
Open Source Hardware

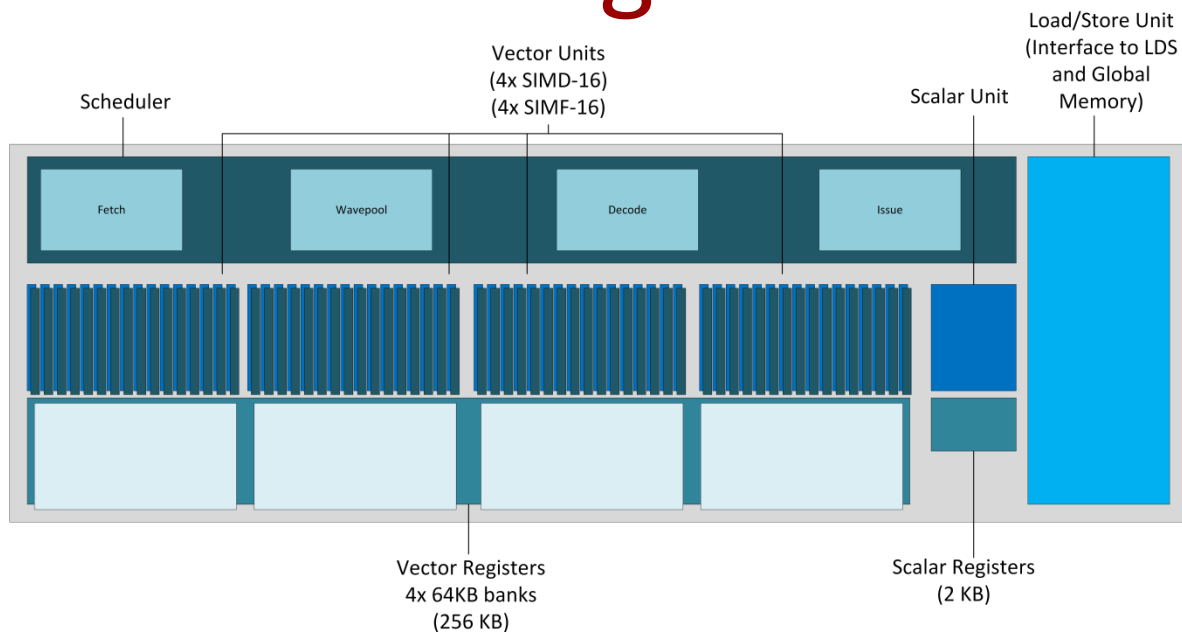


MIAOW vs. AMD Tahiti

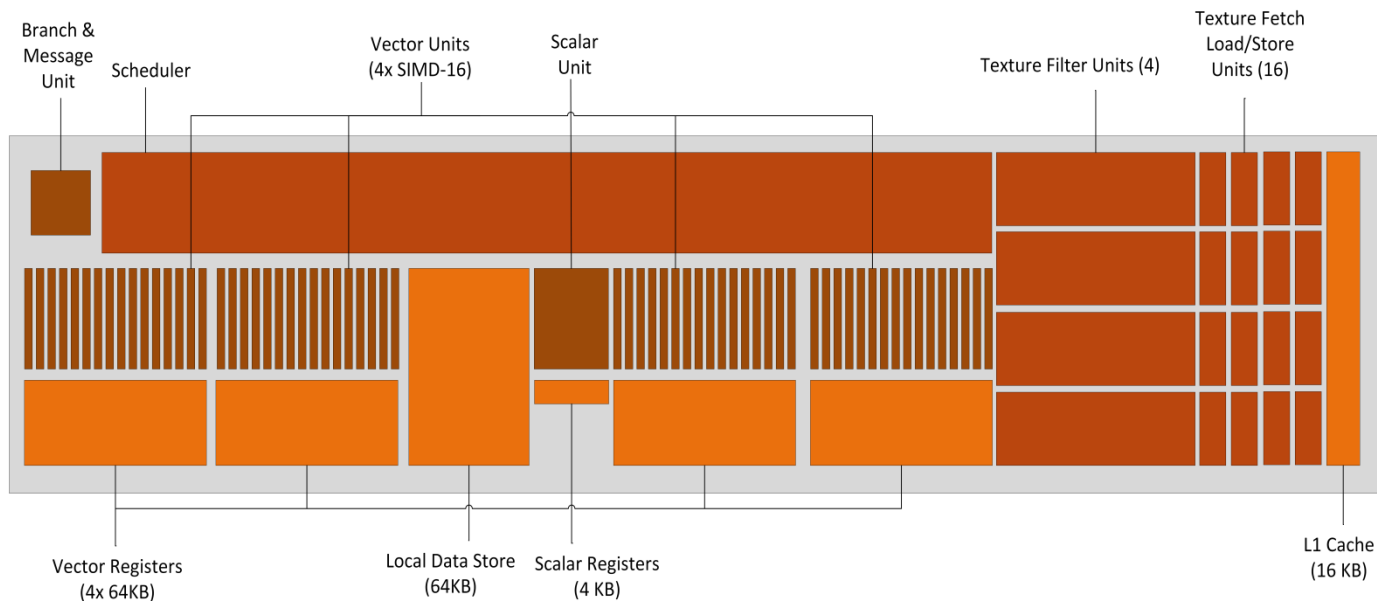


CU Design

MIAOW



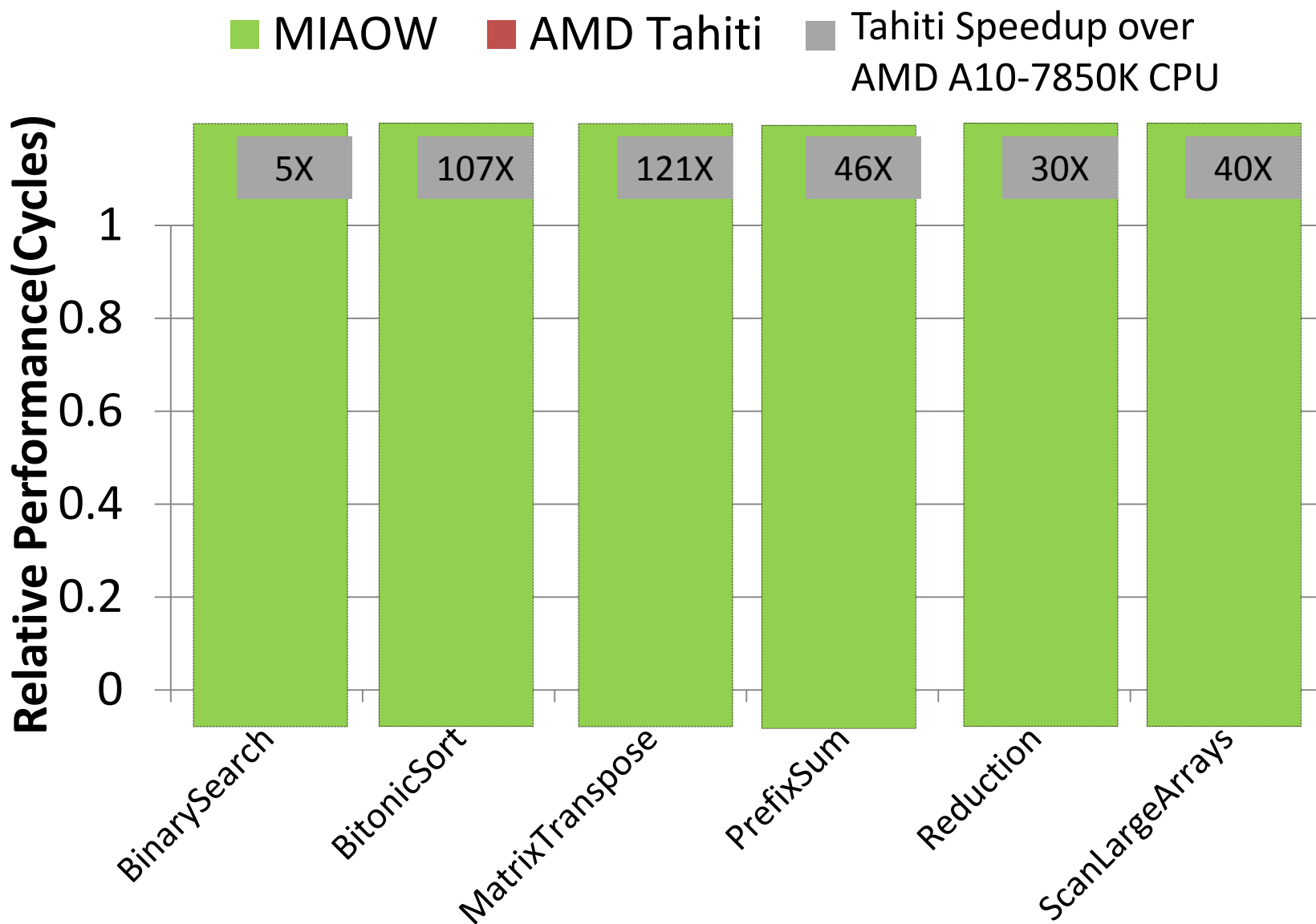
**GCN
CU***



**HOTCHIPS 2014*



Performance Comparison





Area Comparison

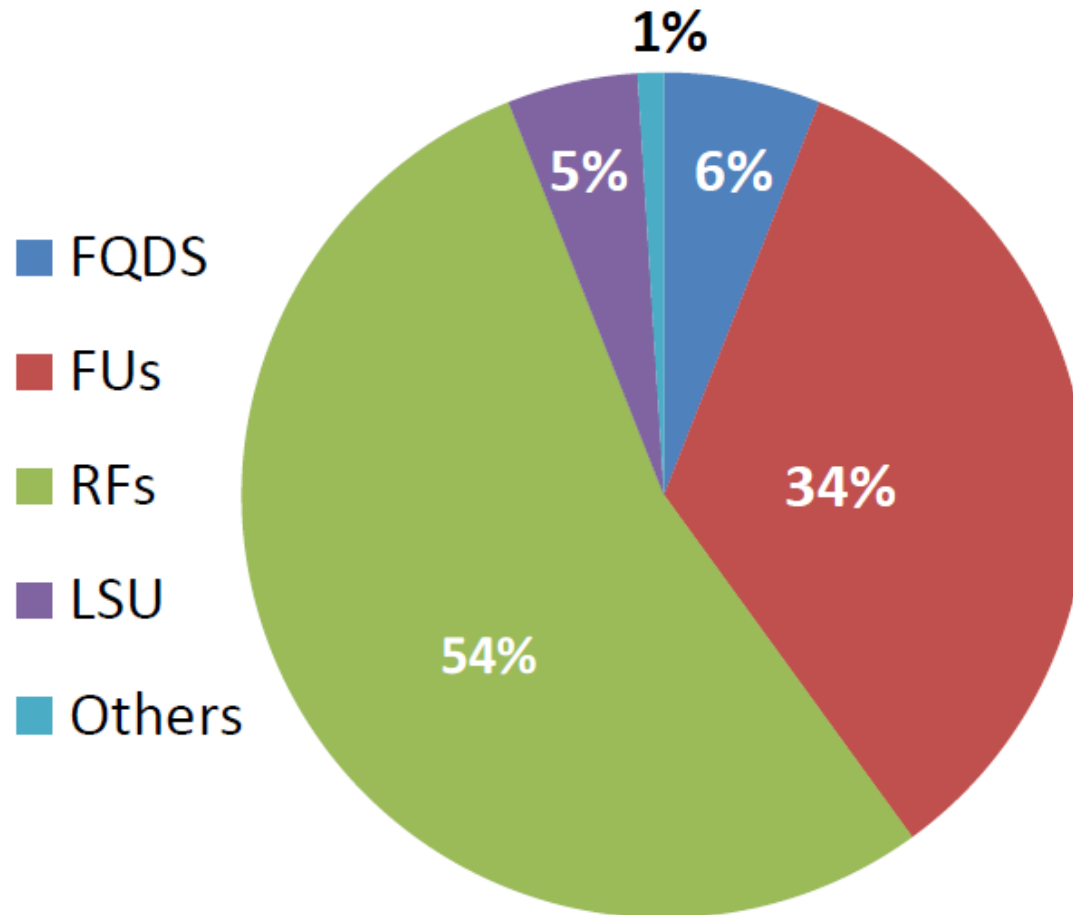
Tahiti CU area*: 5.02 mm² @ 28nm
MIAOW CU area: 9.1mm² @ 32nm



Area Comparison

Tahiti CU area*: 5.02 mm² @ 28nm

MIAOW CU area: 9.1mm² @ 32nm



**Estimate from die-photo analysis and block diagrams from wccftech.com*



Power

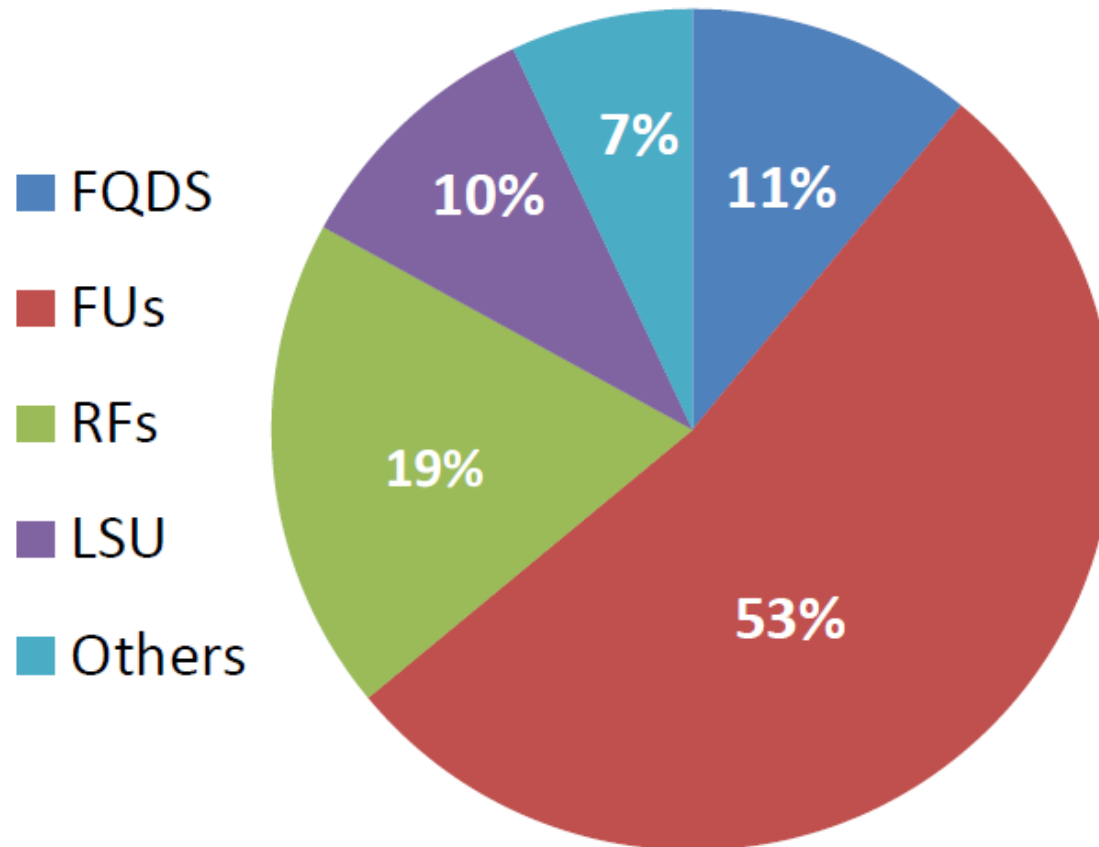
Tahiti CU power*: 0.52 W
MIAOW CU Power: 1.1 W



Power

Tahiti CU power*: 0.52 W

MIAOW CU Power: 1.1 W



* Ballpark estimate from TDP and occupancy



MIAOW is comparable to
industry designs



MIAOW Technical Overview

**Demonstrate MIAOW is a
credible GPGPU**

**Implications and Possibility of
Open Source Hardware**



Lessons Learned

- It was surprising this was doable!
 - Microarchitecture design, HDL implementation, verification was not tedious
- Software toolchain being available was great
- We punted on physical design
- FPGA tools are still quite tedious to use



Implications for Industry

- Open Source Hardware GPU
 - Relevance to OpenCompute & Maker movement
- How can a HW startup benefit from MIAOW
 - Start with MIAOW and focus on innovative pieces from day one
- IP and Compiler
 - License under BSD, ISA is OK, compiler usable
 - How to avoid IP infringement?



Lessons from Open Source S/W

PHP, Linux, ruby, mysql,
sqlite, apache, gcc
late 80s, early 90s

\$0

Facebook, Twitter,
Whatsapp, Instagram
Web 2.0

>\$10 bill.

MIAOW, OpenCores,
RISC-V etc..

\$0

?



What drives Open Source Software?

Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects

- Its fun!
 - “Enjoyment-based intrinsic motivation, namely how creative a person feels when working on the project, is the strongest and most pervasive driver.”
- Its valuable
 - “user need, intellectual stimulation derived from writing code, and improving programming skills are top motivators”



Conclusion

- MIAOW is transformative for GPU research
- Its role in open source hardware movement?
- Are open source hardware chips feasible?
- More community support → First open Source Silicon GPU Chip



www.miaowgpu.org

HOTCHIPS 2015

Many technical details in this publication: TACO 2015: Enabling GPGPU Low-Level Hardware Explorations with MIAOW: An Open-Source RTL Implementation of a GPGPU

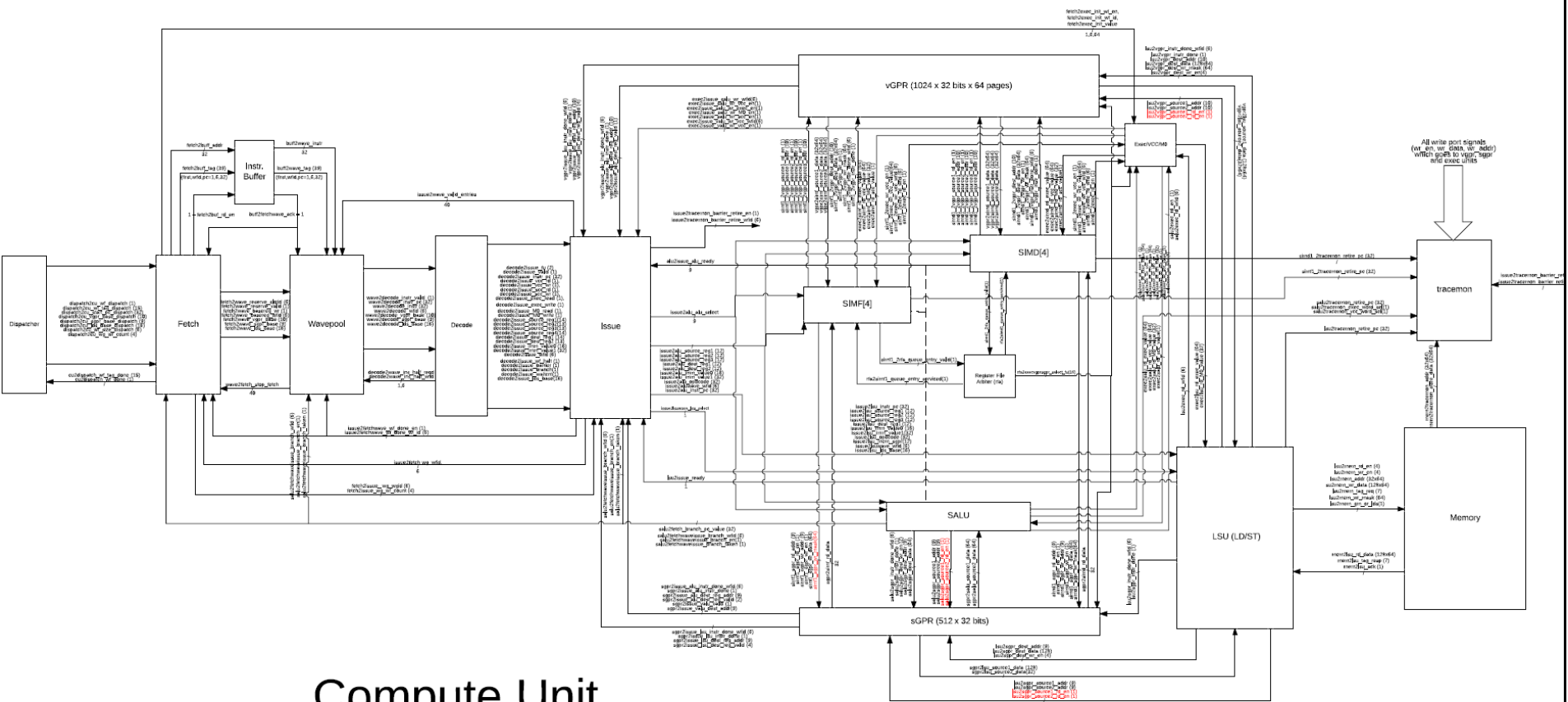




Demo Video



Back Up Slides





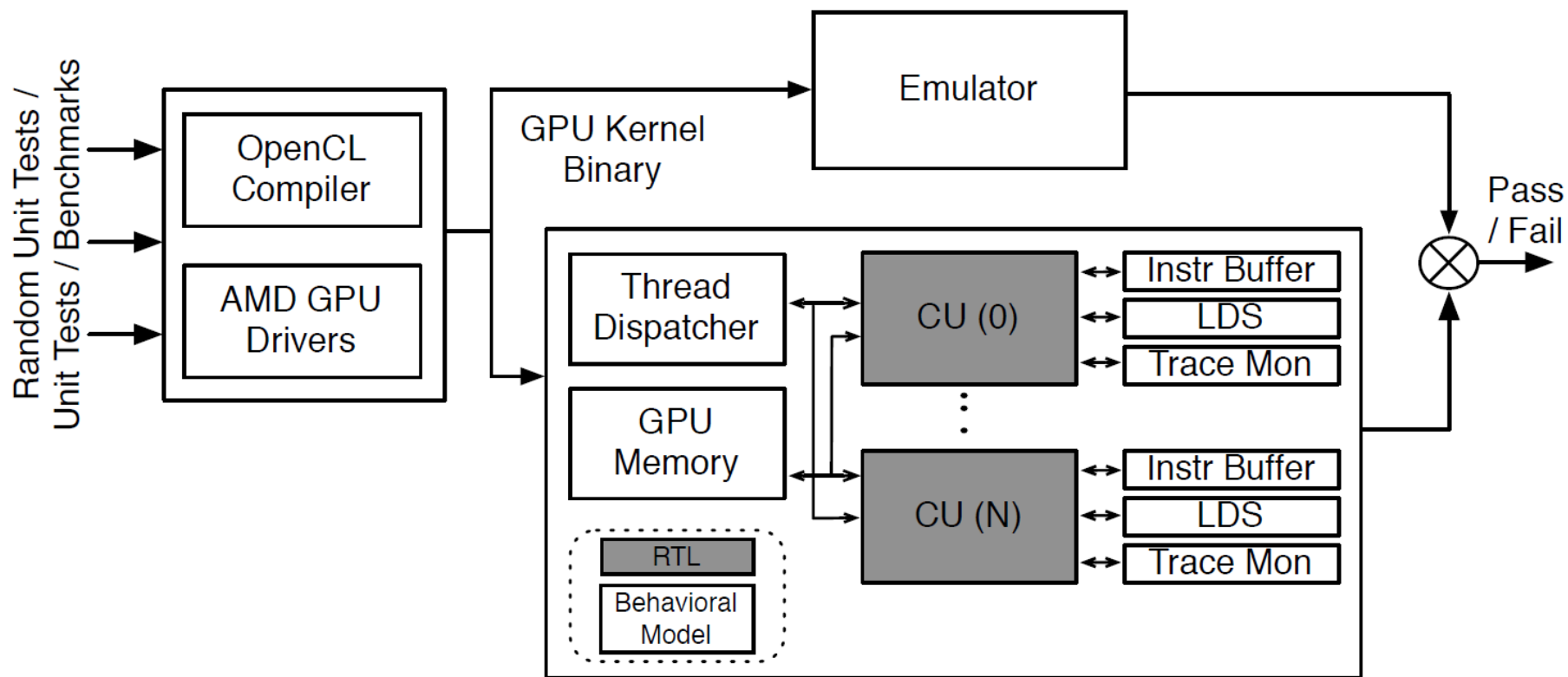
Flexibility

Design choice	Realistic	Flexibility	Area/Power impact
Fetch bandwidth (1)	Balanced [†]	Easy to change	Low
Wavepool slots (6)	Balanced [†]	Parametrized	Low
Issue bandwidth (1)	Balanced [†]	Hard to change	Medium
# int FU (4)	Realistic	Easy to change	High
# FP FU (4)	Realistic	Easy to change	High
Writeback queue (1)	Simplified	Parametrized	Low
RF ports (5,4)	Simplified	Hard to change	High
RF ports (SRAM) (1)	Realistic	Hard to change	Low
Types of FU	Simplified	Easy to change	High

[†]*Fetch optimized for cache-hit, rest sized for balanced machine.
Numbers in parenthesis indicate the design parameters.*



Verification





As a Research Tool

Direction	Research Idea	MIAOW enabled findings
Traditional μ arch	Thread-block compaction (TBC)	<ul style="list-style-type: none">• Implemented TBC in RTL• Significant design complexity• Increase in Critical Path length
New Directions	Circuit-Failure Prediction (Aged SDMR)	<ul style="list-style-type: none">• Implemented entirely in μarch• Works elegantly in GPUs• Small area, power overheads
	Timing Speculation (TS)	<ul style="list-style-type: none">• Quantifies error-rate on GPU• TS framework for future studies
Validation of Simulator studies	Transient Fault Injection	<ul style="list-style-type: none">• RTL Level Fault Injection• More Gray area than CPUs• Silent data corruption seen